PIXEL-WISE SEGMENTATION OF THE BLOOD VESSELS USING RANDOM FORESTS

Branislav Hesko
Doctoral Degree Programme, FEEC BUT
E-mail: xhesko00@stud.fit.vutbr.cz

Supervised by: Vratislav Harabiš
E-mail: harabis@feec.vutbr.cz

Abstract: This paper presents segmentation of the blood vessels in retinal images. First, a serie of feature detectors is applied in form of multiple filters. Then, each pixel is classified using random forests, which was trained on labeled images. Promising results have currently been achieved.

Keywords: segmentation, random forests, retinal blood vessels

1 INTRODUCTION

Segmentation of the blood vessel images is one of the main processes in fundus camera images evaluation. It allows to examinate major eye diseases, such as glaucoma, hypertension or diabetic retinopathy. Automatic segmentation saves time needed to manual segmentation and provides useful informations about diseases without having visible symptoms. Currently many approaches are frequently used, such as supervised or unsuperwised methods with use of machine learning, matched filtering, vessel tracking, deformable models etc. [3], [4], [5], [7]

In this paper we present a novel approach to pixel-wise segmentation. We detected 199 features for each pixel using different filters in form of kernels. Afterwards, classification is realised using random forests. Multiple decision trees are learned and final class is decided by majority voting or averaging. To avoid biasing, features used to build a tree must be uncorrelated, thus randomly sampled.

2 METHODS

In this paper, DRIVE (Digital Retinal Images for Vessel Extraction) database is used. It contains aproximately 30 images with manually segmented images. Each image has resolution of (565×584) pixels. The approach is straightforward. For each pixel a feature map is computed using multiple filters like hessian, roberts diagonal mask, sobel mask, gabor filters and laplacian. The total size of the feature space is 199 features for each pixel.

2.1 FEATURES COMPUTING FILTERS

The first used filter is Roberts diagonal filter, which is computed as a 2D convolution of the image with masks:

$$R_1(x,y) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad R_2(x,y) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{1}$$

These masks represent diagonal approximation of the first derivative in both directions.

Next, 2D convolution with sobel mask is applied, to approximate first derivative in directions x and y.

It is in form:

$$S_x(x,y) = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \text{and} \quad S_y(x,y) = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \tag{2}$$

Another used feature is Laplacian, which is an approximation of the second derivative. It is implemented as a convolution with mask:

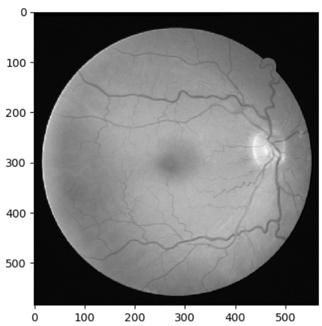
$$L(x,y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{3}$$

Afterwards, Hessian matrix is computed. It is defined as:

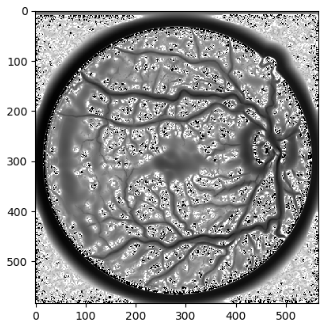
$$H(x,y) = \begin{vmatrix} \frac{\delta^2 I}{\delta x^2} & \frac{\delta^2 I}{\delta x \delta y} \\ \frac{\delta^2 I}{\delta x \delta y} & \frac{\delta^2 I}{\delta y^2} \end{vmatrix}, \tag{4}$$

where I is the original image. Hessian is used to determine extremas of a function $f(x,y)$ and is given by its determinant.

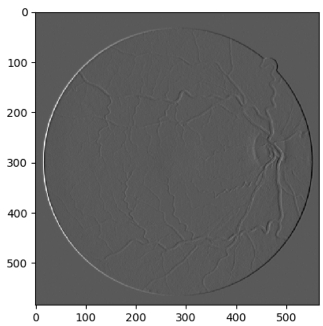
Negative logarithm of the Hessian matrix is taken as a feature. Vessel's pixels are having darker colors. Testing image and its Hessian is shown on the figure 1.



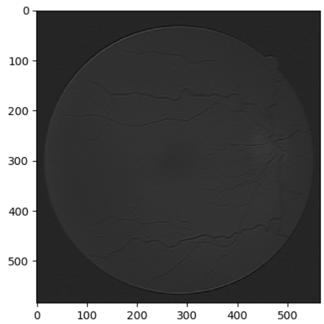
(a) Original image



(b) Hessian of the image



(c) Laplacian of the image



(d) Gabor filtered image

Figure 1: Examples of used features together with original image.

Then, original intensities are used as a separate feature because pixel intensities in the vessel are darker in general.

Finally, the rest of the feature space is computed using Gabor filters [6], which are linear filters used for extracting features. Discrete version of the filter is given by:

$$G(i,j) = B \cdot e^{-\frac{i^2+j^2}{2\sigma^2}} \cdot \cos(2\pi f(i\cos(\theta) + j\sin(\theta)), \tag{5}$$

B is normalizing factor, f is the frequency and θ filter orientation. σ is the size of the region, which is taken in concern. [6]

2.1.1 Parameter importance

Majority of the features are computed using Gabor filters. This type of filter has three different parameters. In general, Gabor filter is expressed as Gaussian kernel modulated with plane sinusoidal wave. There are three tunable parameters. Relating these parameters to blood vessel segmentation, f is the frequency of plane sinusoidal wave. Blood vessel diameter and repeatability of vessels are influenced by choosing different freuqncies. Parameter θ gives orientation of the Gabor filter. Filter response is maximised when the orientation of the blood vessel is equal to θ. Finally, the last parameter σ is the most important parameter to detect different diameters of blood vessels. For different σ, different diameter of the blood vessel has its maximum filter response. According to the literature, Gabor filter is most sensible on blood vessels with diameter $d = \sqrt{\sigma}$ and maximal blood vessel diameter is about $d_{max} = 10$ pixels, thus σ has to be choosen according to this value. [6]

After detecting features, the set of images is split into training and testing set equally. Only pixels laying inside retina are used to train.

2.2 Random forests

Random forests is an example of gradient boosting methods. It combines multiple weak classifiers to obtain one robust classifier. Due to the averaging, random forests does not suffer from overtraining, but produce a limiting value of the generalization error. [2]

There are many variants of the tree building methods, but in general k trees are generated, but unlike bagging, random forests does not construct a tree from all of the features. Instead, trees are constructed usually from $\sqrt{N}$ features, where N is the number of features. Best split feature from the subset of randomly choosen features is used to split each node in a tree based on the training set. The goal is to have all of the k trees uncorrelated, because only in this case, averaging multiple decision trees is less sensitive to noise and does not result in biasing. Random forests are also relatively robust to outliers.

For a testing sample, each decision tree classifies this sample and the result is obtained using averaging of the predictions of all individual decision trees. [2], [1]

3 RANDOM FORESTS SEGMENTATION

The set of the total 199 features is entering into random forest pixel-wise classification. Pixel can be predicted as vessel or background with corresponding values $(0,1)$. Features does not have to be normalized, because random forests are not comparing features mutually, unlike SVMs. The random forest with the following parameters is used:

- Number of decision trees: 100
- Class weights: $(1,3)$ - this means that vessel pixels are three times more relevant in the error estimation.
- Maximum number of features : $\sqrt{N}$, where N is the number of features. This value indicated the number of features used to train each decision tree.
- Maximum depth: All of the decision trees are fully grown, thus all the leafs are pure.
- Bootstraping: Statistical technique, which involves random sampling with replacement [1].

- Entropy: A measure for estimating quality of split, when training a decision tree. Entropy could be expressed as information gain.

At the borders of the background and retina, usually features are causing, that vessel pixels are incorrectly detected. To eliminate this problem, only pixels laying inside retina could be classified as vessel pixels. This is achieved using mask of the retina pixels in the original image, which is part of the dataset and also is easy to detect without dataset because background pixels are having values close to 0.

4 RESULTS

The dataset is split into training and testing sets. Training set contains 66% images, while testing set contains 33% of the resting images. For each pixel, its features are computed and separately classified. This implies that neighborhood is not influencing pixel classification result. But partially, many features are computed using adjascent pixels via 2D convolution. Thus, some information about neighborhood is preserved in features. After segmenting all of the retina pixels, image is obtaining by putting pixels in their corresponding position. Segmented image together with corresponding mask are inserted below:

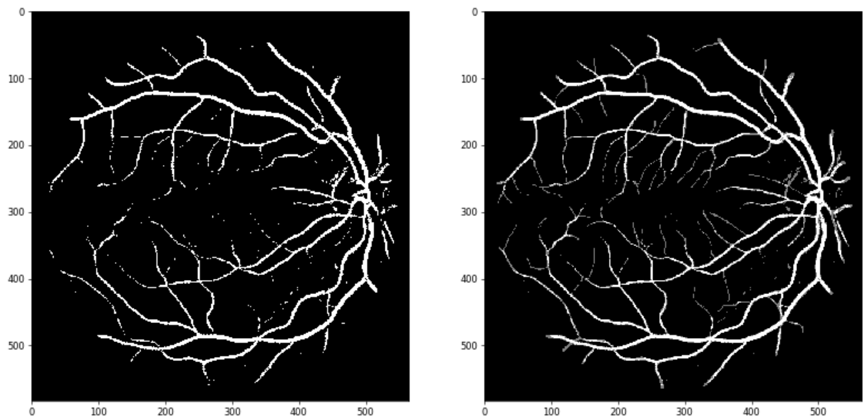


Figure 2: Result of the pixel-wise segmentation - left, target values - right.

For every image, we compute sensitivity, specificity, dice coefficient and intersection over union coefficient. We put results in the following 1:

Based on this table, we can observe high specificity and sensitivity, but smaller values of intersection over union caused by small vessel misclassification and detection of optical disc. Dice coefficient should be ideally equal to 1.

4.1 PERSISTING PROBLEMS

There are three major persisting problems, causing some pixels to be badly classified. First, optical disc is usually segmented as blood vessel. This is probably caused by change in brightness on the optical disc border, which is evaluated the same way as the contrast in blood vessels.

Next, on the border of retina, without removing it according to the mask, Pixels are having feature values close to the vessel values, thus they are classified as blood vessels. By detecting retina on the image, this type of errors should be removed.

Finally, the biggest problem lays in detection of small blood vessels. Their features values are not sufficiently close to the features of bigger vessels, resulting in false negative case classification. This

Table 1: Segmentation result for each image in testing set.

Image	Sensitivity	Specificity	Intersection over union	Dice coefficient
1	0.8750	0.940	0.453	0.722
2	0.856	0.954	0.567	0.734
3	0.861	0.951	0.554	0.740
4	0.852	0.799	0.299	0.549
5	0.913	0.926	0.513	0.782
6	0.888	0.917	0.529	0.725
7	0.912	0.891	0.427	0.725
8	0.873	0.934	0.515	0.723
9	0.884	0.933	0.518	0.737
Average	0.879	0.916	0.486	0.718

problem must be treated in the future, probably by applying another features, like rotated gaussian filters [5]. One of the causes of this problem is also relatively small number of small vessels pixels in the training dataset.

5 CONCLUSION

We have presented an approach to pixel-wise segmentation using multiple features detectors followed by random forest classifier. This approach is succesful on bigger vessels, but often insufficient on smaller vessels. This is also caused by relatively small number of smaller vessels in the image. Moreover, manual segmentation is subjective and may contain false pixels, which are decreasing classification success rate. In the future work, we will employ more feature detectors which are more sensitive to smaller vessels. Then, we have to consider, that DRIVE databse is outdated according to image resolution and quality, thus we will employ our algorithms to a better database, like HRF. Finally, postprocessing may be applied, for example, vessel tracking, which could connect small vessels.

REFERENCES

[1] Efron, B.: Bootstrap methods: Another look at jackknife, The Annals of Statistics (1), 1979, p. 1-26.

[2] Breiman, L.: Random Forest, Machine Learning 45(1), 2001, p. 123-140.

[3] Fraz, M.M. et al: Blood vessel segmentation methodologies in retinal images - A survey, Computer Methods and Programs in Biomedicine, 2018, p. 407-433.

[4] Saffarzadeh, V.M. et al.: Vessel segmentation in retinal images using multi-scale line operator and k-means clustering, Journal of Medical Signals, 2014, p. 122-129.

[5] Singh, N., Kaur, L.: A survey on blood vessel segmentation methods in retinal images, EDCAV, 2015, DOI: 10.1109/EDCAV.2015.7060532

[6] Mehrotra, R.: Gabor filter-based edge detection, Pattern Recognition, 1992, p. 1479-1494, DOI: 10.1016/0031-3203(92)90121-X.

[7] Zhang, J. et al.: Blood vessel segmentation of retinal images based on neural network., Image and Graphics, 2015, 6p.